

Þróun Hugbúnaðar A
Hópverkefni
HBV402G –Vor 2018

Leikjaland

Lokaskýrsla

Árni Þorkels Árnason
Hafðís Mist Bergsteinsdóttir
Kristín Júlía Ásgeirsdóttir
Rebekka Rut Petersen

15. apríl 2017



HÁSKÓLI ÍSLANDS
VERKFRÆÐI- OG NÁTTÚRUVÍSINDASVIÐ

Efnisyfirlit

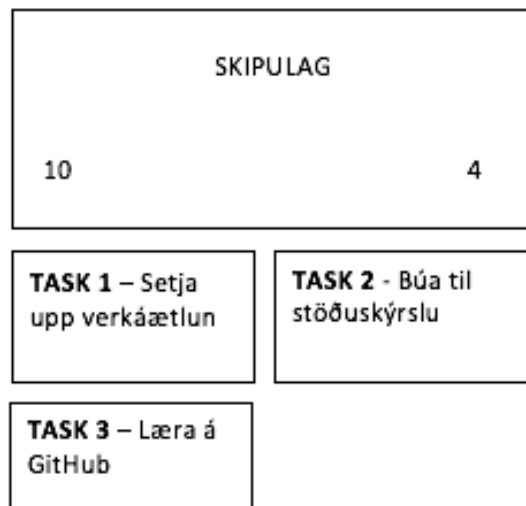
1. Inngangur	3
2. Hönnun, hugbúnaðarkerfi og strúktúr.....	7
3. Núverandi staða verkefnisins	8
4. Næstu skref.....	9
5. Hvað hefur verið lært og áunnist	9
6. Samantekt.....	10
7. Klasa diagröm	11
8. Forritunarkóði	12

1. Inngangur

Leikurinn er hannaður fyrir börn á aldrinum 5-10 ára. Hann samanstendur af einum keppanda og fjórum herbergjum: byrjunarherbergi, skólastofu, útiveru og Leikjalandi. Leikurinn gengur út á það að leikmaður þarf að leysa þrautir í skólastofunni og útiverunni til að komast loks í Leikjaland og eiga möguleikann á því að sigra leikinn.

Leikmaðurinn byrjar í byrjunarherberginu þar sem hann ákveður hvaða kyn hann vill vera. Eftir það er spurt hann hvort hann vilji fara út að leika eða í skólann. Til að fara í skólann velur leikmaðurinn skólaföt og til að fara út að leika velur hann útiföt. Þá er leikmaðurinn kominn í næsta herbergi, þ.e. annað hvort í skólann eða út. Í skólanum þarf leikmaðurinn að leysa þrjú stærðfræðidæmi, til að komast út úr því herbergi og yfir í það næsta. Til að sanna að hann hafi klárað öll dæmin þá fær hann að launum skólabók en hann þarf að hafa hana til að komast í lokaherbergið. Eftir að hafa leyst stærðfræðidæmin fer leikmaðurinn út að leika. Þar þarf hann að leysa þrjár myndaþrautir. Þegar hann hefur náð því fær hann blóm að launum. Nú er hann búin að vinna sér inn bæði skólabókina og blómin og kemst því inn í Leikjaland en leikmaðurinn þarf að hafa báða hlutina meðferðis til að komast inn í lokaherbergið. Í Leikjalandi bíða hans nokkrir mótspilarar en leikmaðurinn velur einn af þeim til að spila við sig leik sem gengur út á það að giska á töluna sem mótspilarinn velur. Þegar leikmaðurinn hefur náð því hefur hann sigrað leikinn!

Helstu kröfur verkefnisins voru að læra á Python og skilja grundvallaratriðin, t.d. læra að búa til notendasögur og UML rit, skilja muninn á milli falls og klasa og kunna að vinna með helstu skipanir sem Python notar. Leikurinn var hannaður í fjórum sprettum þar sem notendasögur voru skilgreindar í byrjun með bestu getu en þær voru 10 talsins. Hér fyrir neðan má sjá þær vera brotnar niður í smærri einingar (tasks).



KEPPANDI

10

3

TASK 1 - Velja kyn og fatnað (annað hvort skóla- eða útiföt)

HERBERGI 1 BYRJUNARREITUR

10

3

TASK 1 - Tengjast öðrum herbergjum (annað hvort skólastofu eða útiveru)

HERBERGI 2 KENNSLUSTOFA

15

5

TASK 1 - Inni í kennslustofunni er kennari sem hjálpar keppandanum

TASK 2 - Útbúa auðveld stærðfræðidæmi sem keppandinn þarf að leysa

TASK 3 - Ef keppandinn nær að leysa stærðfræðidæmin þá fær hann skólabók að launum og kemst inn í næsta herbergi

TASK 4 - Ef keppandinn hefur ekki leyst myndaþrautirnar í útiverunni (herbergi 3) þá þarf hann að fara aftur í herbergi 1 og skipta í útiföt svo hann komist út.

TASK 5 - Ef keppandinn hefur bæði leyst stærðfræðidæmin og myndaþrautirnar í báðum herbergjunum þá kemst hann inn í næsta herbergi

HERBERGI 3 ÚTIVERA

15

5

TASK 1 - Þegar keppandinn fer út bíður hans vinur sem hjálpar honum

TASK 2 - Útbúa þrjár myndaþrautir sem keppandinn þarf að leysa

TASK 3 - Ef keppandinn nær að leysa myndaþrautirnar þá fær hann blóm að launum og kemst inn í næsta herbergi

TASK 4 - Ef keppandinn hefur ekki leyst stærðfræðidæmin í skólastofunni (herbergi 2) þá þarf hann að fara aftur í herbergi 1 og skipta í skólaföt svo hann komist í skólastofuna

TASK 5 - Ef keppandinn hefur bæði leyst stærðfræðidæmin og myndaþrautirnar í báðum herbergjunum þá kemst hann inn í næsta herbergi

HERBERGI 4
LEIKFANGALAND

20

3

TASK 1 - Útbúa
teningapraut
sem keppandinn
þarf að leysa til
að vinna leikinn

TASK 2 - Ef
keppandinn vinnur
þá kemur loklag

INNTÖKUSKILYRÐI
til að komast í Leikfangalandið

30

3

TASK 1 -
Skólabók úr
skólastofu

TASK 2 - Fimm gul
blóm úr
blómabeðinu úti

TASK 3 - Hleypa keppanda inn í
lokaherbergið ef task 1 og task 2 er
uppfyllt

TENGING MILLI BORÐA/HERBERGJA

30

5

TASK 1 - Ef
keppandi er í
réttum fötum þá
kemst hann inn í
næsta herbergi

TASK 2 - Borð 1
(byrjunarreitur)
hefur tengingu við
bæði kennslustofu
og útiveru

TASK 3 - Úr bæði útiveru og kennslustofu
hefur keppandinn val um að reyna að
komast inn í lokaherbergið ef hann hefur
uppfyllt öll inntökuskilyrðin

ÞRAUTIR

25-40

3

TASK 1 - Borð 2

TASK 2 - Borð 3

TASK 3 - Skila
„sönnunum“
fyrir kláruð borð

TASK 4 - Borð 4
(Leikfangaland)

HLJÓÐ OG MYND

50

5

TASK 1 - Fyrir
keppanda

TASK 2 - Fyrir
herbergin

TASK 3 - Hljóð fyrir samskipti ef
keppandinn kemst úr einu borði í annað

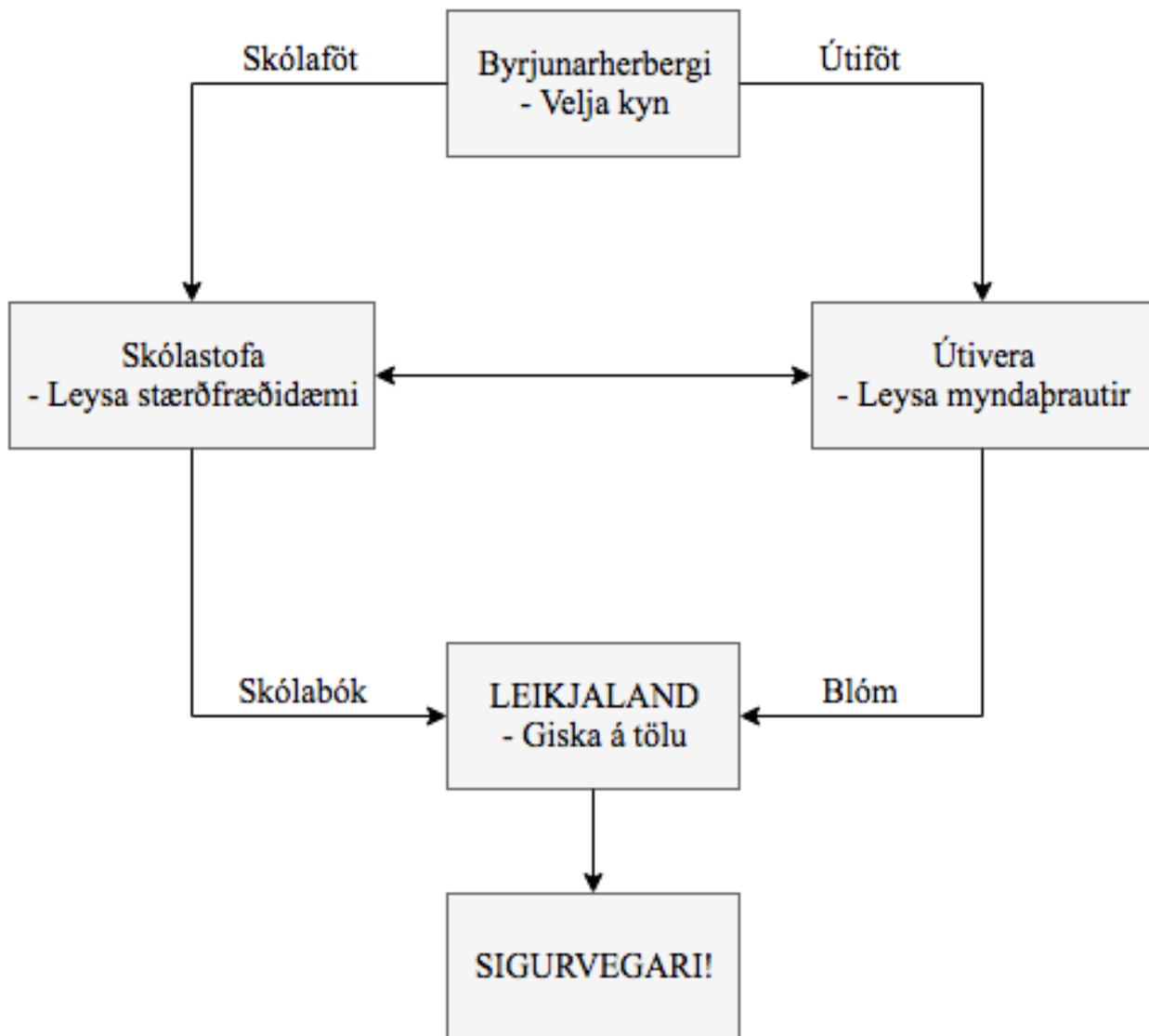
TASK 4 - Borð 4, loklag fyrir
keppandann ef hann vinnur

Notendasögurnar voru skiptar niður á milli sprettanna en hér fyrir neðan má sjá hvaða notendasögur voru gerðar í hverjum spretti fyrir sig. Eins og sjá má var áætlað að hljóð og myndir yrði sett inn í sprett 5. En þar sem kunnátta hópsins jókst við gerð verkefnisins náðist að setja það inn í spretti 3 og 4 og er það því hluti af leiknum eins og hann er núna.

Sprettur 1	Sprettur 2	Sprettur 3	Sprettur 4	Sprettur 5
Skipulag	Byrjunarreitur	Herbergi 2	Herbergi 4	Hljóð og myndir
	Keppandi	Herbergi 3	Tenging á milli herbergja	Aðrar þrautir
		Þrautir fyrir herbergi 2 & 3	Inntökuskilyrði	

Nokkur upphafleg markmið leiksins hafa breyst. Til að mynda átti leikurinn að heita Leikfangaland en breyttist síðar í Leikjaland. Einnig var upprunalega ákveðið að í útiveru herberginu yrði þrautin þannig að það þyrfti að týna ákveðinn fjölda af blómum úr blómabeði til að komast í næsta borð en það þótti of flókið svo ákveðið var að útbúa þrjár myndaprautir sem þarf að leysa. Meginmarkmið leiksins eru þó ennþá eins. Í honum eru fjögur herbergi eins og upprunalega var ákveðið og í þeim eru þrautir eða spurningar sem þarf að svara.

Eitt helsta markmið hópsins við gerð leiksins var að láta hönnunina snúast um að læra á Python og skilja grunnatriðin. Það var því ákveðið að hafa forritið ekki of flókið til að byrja með en eftir að grunnkunnáttan í Python varð meiri var hægt að útbúa flóknari forrit og gera leikinn skemmtilegri. Á næstu blaðsíðu má sjá skýringarmynd sem lýsir leiknum og samskipta á milli herbergja.



2. Hönnun, hugbúnaðarkerfi og strúktúr

Við hönnun á leiknum voru notaðir fjórir klasar, sitthvor klasinn fyrir herbergi 2 og 3, einn fyrir lokaherbergið og einn sem tengir herbergin saman og heldur utan um leikinn. Samskiptin milli klasanna fer því fram að mestu leyti í „Leikjaland“ klasanum (sjá UML rit í kafla 7). Eins og áður hefur komið fram þarf að standast þrautir í hverju herbergi og fá hluti í verðlaun sem eru einskonar inntökuskilyrði til að komast á milli herbergja. Það var hannað með því að nota boolean breytu. Hún segir til um það hvort keppandi sé með hlutinn - ef hann er með hann (TRUE), þá hefur hann náð að standast þrautirnar og ef ekki (FALSE), þá þarf hann að reyna aftur til þess að geta farið á milli herbergja.

Til hliðsjónar við hönnun á leiknum var til að byrja með notaður leikur úr kennslustund sem byggir á klasa og föllum undir honum. Þegar grunnurinn var kominn var notast við Python Libraries en það er einskonar bókasafn yfir innbyggða eiginlega í Python sem auðvelda forritunarkóða. Einnig voru ytri einingar notaðar þegar unnið var með grafík. Til að setja inn myndir var notað graphics.py og til að setja inn hljóð var notað pygames.py en þessir pakkar eru í bókasafni Python og eru sérstaklega hannaðir fyrir einfalda hlutbundna grafík sem hentar vel fyrir Leikjaland.

3. Núverandi staða verkefnisins

Hópurinn hefur verið mjög virkur í verkefninu og eytt miklum tíma í leikinn. Álagið hefur dreifst nokkuð jafnt yfir hópmeðlimi. Í upphafi á hönnun leiksins var hugsað um þrautir fyrir hvert herbergi og hversu langan tíma það gæti tekið að búa til hvert borð fyrir sig. Búið var til notendasögur fyrir hvert herbergi út frá því eftir bestu getu. Þegar leið á önnina komu upp fleiri hugmyndir og leikurinn þróaðist í samræmi við það. Notendasögurnar breyttust því lítillega og tímaáætlun fór úr skorðum eins og búast má við í verkefni eins og þessu. Það var útbúið stöðuskýrslu og verkáætlun fyrir alla sprettina sem hélt utan um nýjar hugmyndir og uppfærslur.

Í fyrsta sprettinum var lagður grunnur fyrir heildarmynd verkefnisins og gert gróft plan með notendasögum fyrir næstu spretti. Í spretti tvö var gerður kóði fyrir fyrsta herbergið og gróf tímaáætlun. Einnig var útbúið UML rit (sjá kafla 7) og burn down rate graf. Sprettur þrjú fór mest í að klára að forrita fyrstu þrjú herbergin, tengja þau saman og bæta við grafík (aðallega myndum). Það var einnig uppfært UML ritið og búið til nýtt burn down rate graf en á þessum tímamarki innihélt leikurinn aðeins einn klasa og sex föll. Í fjórða sprettinum var forritað lokaherbergið ásamt því að bæta við grafík og hljóði. Samskiptin á milli herbergjanna voru betrubætt og útbúnir voru fleiri klasar og tilheyrandi föll undir þeim. Að lokum var farið yfir mögulegar villur til að fá heildarmynd leiksins og útbúið kynningu.

Mögulegar villur sem hefðu getað komið upp við gerð leiksins voru greindar með áhættugreiningu. Til dæmis ef ekki hefði náðst að mynda tengingu á milli herbergja þá hefði leikurinn ekki gengið upp. Auk þess ef tímaáætlunin hefði verið óraunhæf og hópurinn hefði byrjað að vinna í verkefninu of seint þá hefði ekki verið hægt að klára leikinn fyrir lokaskil. Önnur villa sem hefði getað komið upp væri að grafík og hljóð hefði farið úrskeiðis eða að myndirnar hefðu ekki getað opnast í öllum tölvum. Einnig var möguleiki á því að eiga í erfiðleikum með tengingu milli tölva á GitHub. Hópurinn kom saman og ræddi um mögulegar villur og meðhöndlun þeirra ásamt afleiðingum sem það gæti haft á verkefnið. Áhættumatið

skilaði góðum árangri og það hjálpaði að sjá hvað hópmeðlimar óttast að muni eiga sér stað og þá var auðveldara að fyrirbyggja að áhættan myndi ekki eiga sér stað.

Framkvæmdar voru kerfisprófanir sem gengu út á það að print() skipanir voru settar inn í öll föll með talningu (print('test#1'), print('test#2') o.s.frv). Með því var haldið utan um í hvaða föll forritið keyrði og fundið þannig mögulegar villur. Leikurinn var prófaður af hópmeðlimum og utanaðkomandi fólki. Fengin var 6 ára gömul stelpa til þess að spila leikinn og fannst henni leikurinn skemmtilegur og þrautirnar miskrefjandi. Ekki var nægur tími til þess að útbúa unittest prófanir þar sem það var ekki gert fyrir sprett 3 og því voru print() skipanirnar látnar nægja og reyndist það vel. Þegar það kom í ljós að forritið væri að keyra í rétt föll með ofangreindum prófunum voru print() skipanirnar settar í athugasemd svo þær myndu ekki prentast út þegar leikmaður spilar leikinn.

Eins og áður hefur komið fram ákvað hópurinn í upphafi verkefnisins að hafa leikinn frekar einfaldan. Það var því ekki gert ráð fyrir mikilli grafík og hljóði en yfir tímann þróaðist verkefnið, kunnátta hópsins jókst og útbúin var mun meiri grafík en áætlað var. Með þessu sést því að það má alltaf bæta og gera betur með aukinni æfingu. Í framtíðinni væri hægt að bæta við enn meiri og flottari grafík og hljóðum á ýmsum stöðum til að hafa leikinn faglegri og flottari.

4. Næstu skref

Markmiðið er að halda áfram að þróa leikinn og gera hann ennþá betri. Næsta skref er að framkvæma takka sem notandinn ýtir á þegar hann svarar spurningunum í leiknum í staðinn fyrir að þurfa að skrifa inn svörin í skipanaglugga. Þar sem leikurinn er hannaður fyrir ung börn auðveldar það þeim að spila leikinn. Langtímamarkmið er að fá inn aldur hjá notandanum og geta útfært leikinn fyrir mismunandi aldurshópa. Þá geta börn á aldrinum 4-6 ára fengið auðveldari spurningar heldur en þau sem eru t.d. 10-12 ára og leikurinn gæti þá höfðað til stærri markhóps. Framtíðarmarkmið eru einnig að útbúa unittest prófanir og þá er hægt að hafa ennþá flóknari eiginleika í leiknum.

5. Hvað hefur verið lært og áunnist

Helstu vandamálin sem upp komu við gerð leiksins voru að útbúa UML klasaritinn þar sem þau vinnubrögð voru ný fyrir hópinn. Um leið og kunnáttan jókst og forritið fór að tengjast betur saman með fleiri klösum og föllum einfaldaði það gerð leiksins til muna. Í upprunalega leikjaplaninu var áætlunin að hafa einn klasa og mörg föll undir honum en þegar var verið að tengja leikinn saman sást að það var hentugra að nota fleiri klasa. Upp komu vandamál þegar

forritinu var breytt en með hjálp kennara og UML ritinu gekk það að lokum og áhættan varð því ekki mikil. Helstu vandamál sem upp komu í spretti 4 snérust að hönnun leiksins. Eitt vandamálið lýsti sér þannig að þegar leikmaðurinn gerði villu í þraut var hann boðinn velkominn í sama herbergi í stað þess að fá að spreyta sig aftur á þrautinni. Vandamálið var þó leyst með því að bæta við öðru falli efst í klasana sem viðkemur einungis því að bjóða leikmanninum velkominn í það herbergi sem hann er í. Leikmaðurinn getur nú reynt við þrautirnar eins oft og nauðsynlegt er án þess að vera boðinn velkominn í hvert sinn.

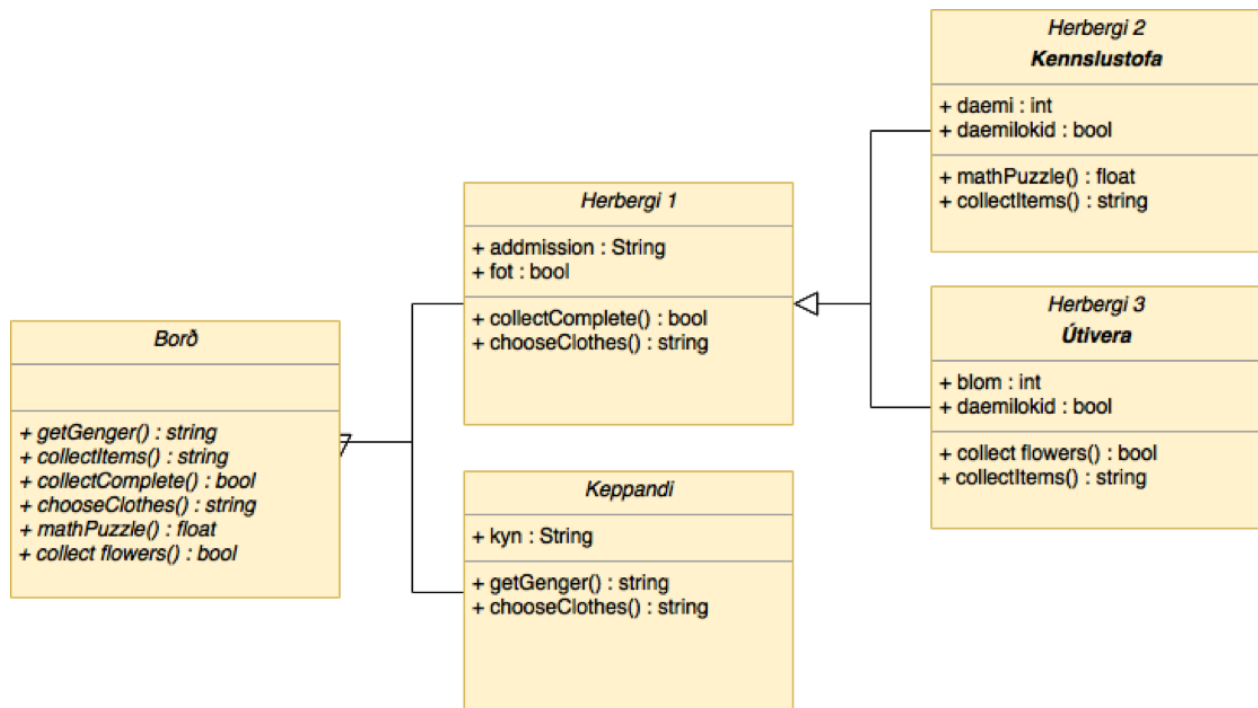
6. Samantekt

Í heildina litið gekk verkefnið vel fyrir sig og kom betur út en þorað var að vona. Verkefnið var krefjandi og tímafrekt en á sama tíma mjög lærdómsríkt og skemmtilegt. Kennarinn tók virkan þátt við gerð leiksins og var samstarfið gott sem hjálpaði til þar sem hópmeðlimir voru óreyndir í tölvuleikjagerð og námsefnið þungt. Í byrjun gekk vel að koma hugmyndum á blað og komast af stað en allir sprettirnir voru þó heldur krefjandi. Með góðri skipulagningu og virkni frá öllum hópmeðlimum þá náðist að klára sprettina með prýði. Þó komu upp nokkrir hnökrar, sérstaklega með kóðann. Það var t.d. erfitt að tengja saman herbergin en það gekk að lokum. Helsta vandamálið var þó að læra að gera UML ritin. Hópurinn var ekki alveg með á hreinu fyrst um sinn hvernig ætti að útbúa slíkt rit þar sem skilningurinn var takmarkaður. En í lok verkefnisins kom í ljós að betra er að hafa fleiri klasa og færri föll undir þeim og tengja UML við klasana þar sem kóðinn var orðinn heldur langur. Hópstarfið gekk mjög vel og tóku allir meðlimir virkan þátt í gerð leiksins og skýrslanna. Engar alvarlegar áhættur áttu sér stað og villumeðhöndlun var framkvæmd fagmannlega.

Áfanginn var mjög umfangsmikill en kennarinn náði að koma efninu ágætlega frá sér miðað við flækjustig verkefna fyrir nemendur á byrjendastigi í forritun. Þar sem áfanginn er mjög þverfaglegur og fer grunnt í mörg atriði hefur áunnist mikill lærdómur á því að hanna tölvuleik sem slíkan. Þegar umfangsmikið verkefni sem slíkt er sett fyrir í áfanga er mikilvægt að byrja sem fyrst á verkefninu og skipuleggja sig vel. Einnig er lykilatriði að vinna jafnt og þétt í hóp yfir allan tímann sem er gefinn til að verkefnið gangi upp. Best er að setjast niður í byrjun verkefnisins, áætla tíma og ekki ætlast til of mikils. Verkefnið þróast síðan með tímanum og nemendur öðlast meiri þekkingu.

7. Klasa diagröm

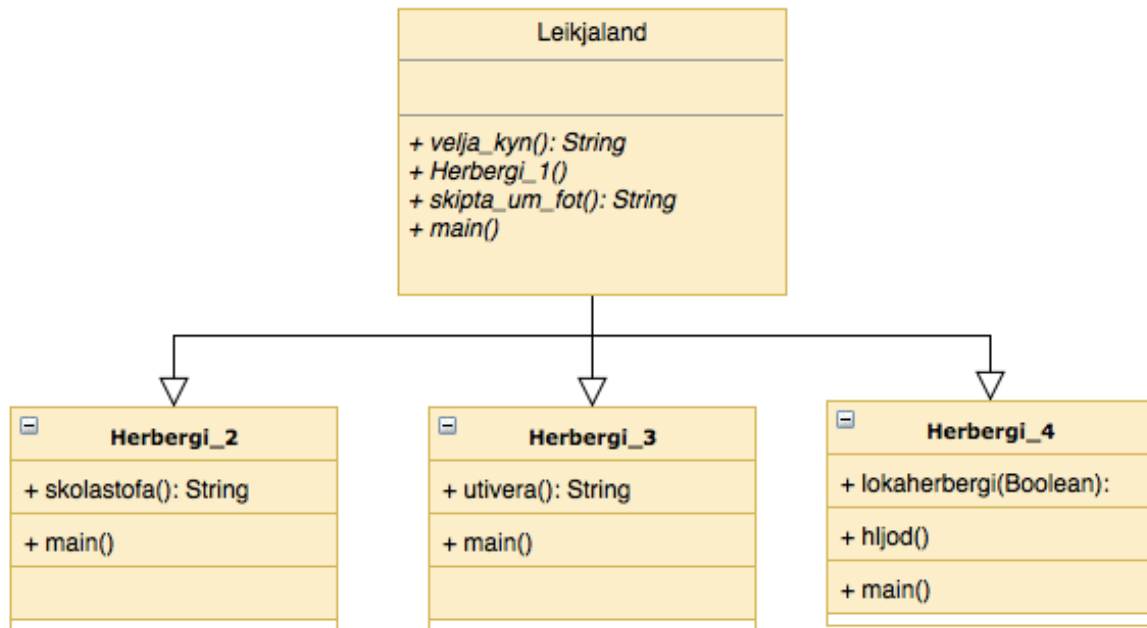
Hér fyrir neðan má sjá klararitin sem voru gerð fyrir hvern sprett.



UML rit 1: fyrir sprett 2



UML rit 2: fyrir sprett 3



UML rit 3: fyrir sprett 4 – eins og það er núna

Eins og sjá má hafa klasaritin uppfærst í hverjum spretti. Ástæðan fyrir því er sú að þegar sprettur tvö eða hlið 2 lokaðist var ekki komin lokamynd á forritunarkóðann. Því var UML ritið ekki í takt við forritið í spretti þrjú og var það því uppfært til að hafa tengingu við forritunarkóðann en hann innihélt þá aðeins einn klasa. Í spretti 4 var bætt við fleiri klösum og uppfært UML ritið í samræmi við það. UML rit 3 sýnir því hvernig það er eins og staðan er núna. Tengingin við UML klasaritin hafa hjálpað mikið til við forritunarkóðann og einfaldað samskiptin á milli herbergja og umfang leiksins. Þegar margir vinna við sama forrit er nauðsynlegt að geta skoðað heildarmyndina.

8. Forritunarkóði

Eins og áður hefur komið fram eru klasar í leiknum með nokkuð einföldum föllum. Forritið er læsilegt og flækjustigið í lágmarki. Leikinn sjálfann er að finna á github undir slóðinni: <https://github.com/Leikjaland/Sprint-4>